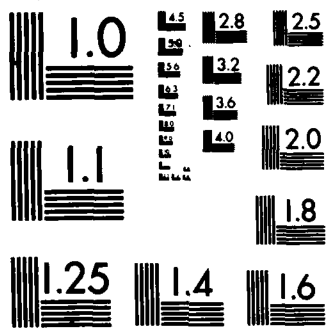
AD-A173 574   AN EXPERT SYSTEM FOR THE OPTIMAL MESH DESIGN IN THE HP-VERSION OF THE FIN (U) MARYLAND UNIV COLLEGE PARK INST FOR PHYSICAL SCIENCE AND TECH   E RANK ET AL   1/1

UNCLASSIFIED   JUN 86 BN-1053 N00014-85-K-0169   F/G 9/2   NL



ICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A173 574

INSTITUTE FOR PHYSICAL SCIENCE
AND TECHNOLOGY

(15)

Technical Note BN-1053

AN EXPERT SYSTEM FOR THE OPTIMAL MESH DESIGN IN THE HP-VERSION OF THE FINITE ELEMENT METHOD

by

Ernst Rank and Ivo Babuška
Institute for Physical Science and Technology
University of Maryland, College Park, Maryland 20742
U.S.A.

June 1986

UNIVERSITY OF MARYLAND

Encl (1)

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Note BN-1053	2. GOVT ACCESSION NO. AD A173 574	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Expert System for the Optimal Mesh Design in the HP-Version of the Finite Element Method		5. TYPE OF REPORT & PERIOD COVERED Final life of the contract
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Ernst Rank and I. Babuska (DAAD - German Academic Exchange Service)		8. CONTRACT OR GRANT NUMBER(s) ONR N00014-85-K-0169 312/402/568/6 - Grant No.
9. PERFORMING ORGANIZATION NAME AND ADDRESS Institute for Physical Science and Technology University of Maryland College Park, Maryland 20742		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Department of Naval Research Office of Naval Research Arlington, VA 22217		12. REPORT DATE June 1986
		13. NUMBER OF PAGES 31
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release: distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

expert systems, finite element method, mesh design, hp-extension, elasticity, adaptive methods, error estimation.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This paper suggests a simple expert system frame and provides the domain knowledge for the optimal mesh design and the prediction of the error in energy norm for the problem of plane elasticity using the hp-extension in the finite element method. The expert system monitors the progress of the analysis, guides the user through the various steps and is able to reason about its own advise. In an example the user-expert communication is shown and the superiority of the results is demonstrated.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AN EXPERT SYSTEM FOR THE OPTIMAL MESH DESIGN IN THE HP-VERSION OF THE FINITE ELEMENT METHOD (†)

Ernst Rank (*)

Ivo Babuska (**)

Institute for Physical Science and Technology

University of Maryland

(*) Supported by DAAD (German Academic Exchange Service) , No. 312/402/568/6

(**) Partially supported by the Office of Naval Research, Contract N00014-85-0169

(†) Presented at the First World Congress on Computational Mechanics, Austin, Texas, September 86

ABSTRACT

This paper suggests a simple expert system frame and provides the domain knowledge for the optimal mesh design and the prediction of the error in energy norm for the problem of plane elasticity using the hp-extension in the finite element method . The expert system monitors the progress of the analysis, guides the user through the various steps and is able to reason about its own advise. In an example the user-expert communication is shown and the superiority of the results is demonstrated.

Keywords: expert systems, finite element method, mesh design, hp-extension, elasticity, adaptive methods, error estimation.

INTRODUCTION

In recent years, the p-version of the finite element method has attracted considerable interest , as it has been shown theoretically and practically, that higher accuracy at lower cost and better rates of convergence can be obtained as in the h-version of the FEM.

However, the performance of h- and p-version hinge on a proper mesh design especially in the neighborhood of singular points and lines in the exact solution of the problem. Whereas in the h-version adaptive finite element codes are already available , which construct optimal meshes with respect to the minimization of some norm of the approximation error, only few results are known for an optimal mesh design in the p-version . Yet, it has been shown theoretically and numerically for some model-problems, that a proper mesh-refinement combined with an increase of the polynomial degree , i.e. an *hp-version* of the FEM, leads to exponential rate of convergence and superior accuracy.

Usually it is impossible to construct *a priori* the optimal relation of polynomial degree and finite element mesh in practical cases, where often dozens of 'critical areas', i.e. regions around singularities or around strongly curved boundaries occur.

This paper suggests an *expert system*, which uses *a posteriori* information from a cheap starting computation for a proper mesh design in the hp-version. The goals are the following :

- The input data of the user should be kept to a minimum;

- The user should get rational support for his decisions about the mesh design;

- The system should be able to monitor the progress of the analysis;

- It should advise the user at each time about the next steps to be taken;

- The system should be able do reason about its own advise;

- The expert system frame should be seperated in a problem-independent part , the 'inference engine' and the problem dependent 'knowledge base';

- A user should be able to get superior results with the help of the expert system, i.e. highest accuracy should be achieved for given cost;

The tools of the expert system are , among others, a p-version finite element code and an interactive graphical mesh generator.

Numerical examples show, that by this approach an exponential rate of convergence and

superior accuracy at low computational cost and storage is obtained in practical problems with typically 10 to 20 singular points, i.e. stress concentrations at reentrant corners, crack tips or change of boundary conditions.

In the next chapter we will give an outline of the basic ideas of an expert system in CAD applications. Then we will define the rules for an optimal mesh design in the hp-version. In the final section we will show an example of the application of the system and an appendix will summarize the mathematical foundation of the expert system's rules.

A FINITE-ELEMENT-EXPERT-SYSTEM

The basic concept

Expert systems are being considered one of the most promising developments in modern computer science. Up to now they have mostly been applied in areas, where no clearly defined algorithmic strategies are available,such as prediction, interpretation, diagnosis and monitoring. For an excellent introduction into knowledge based expert systems we refer to /1/.

Knowledge based expert systems are expected to be successfully applicable if (see /1/)

- experts are able to solve problems considerably better than novices

- rules for a successful problem solution can be defined

- heuristics are essential to solve the problem

- the problem domain is of limitted complexity.

Knowledge based systems in engineering analysis and especially computer aided design have been adressed only recently (see e.g. /2/). Yet there seems to be a wide range of possible applications as highly sophisticated computer programs are available, most of them needing a considerable amount of expertise, if used in a reasonable way.

Expert systems usually consist of problem independent *inference engines*, and problem dependent *knowledge bases*. Frequently the knowledge in a field can be separated into two parts (see /3/). *Surface knowledge* in the context of CAD applications consists, for example, of rules such as which algorithm to use when and with what kind of input data. This part

of the knowledge is well suited to be handled by *rule-based expert systems*. *Deep knowledge*, on the other hand, consists of the knowledge about the underlying problem structure. For example, in our context of an optimal mesh design it consists of mathematical rules, which permit to predict together with heuristics the error of a finite element computation in certain norms.

Before we elaborate on an expert system for the optimal mesh design in the hp-version of the finite element method, some general remarks on what an expert system in CAD should do have to be made. Let us first focus on how a *human expert* is expected to help a user in the application of sophisticated computer programs, e.g. a finite element program.

A novice user of the program should be able to ask the expert at each state of the analysis, what he has to do next. The expert has to be able to *guide* the user through an analysis, he has to have the thorough *knowledge* about the application of the program.

The expert has to know at each stage, which subtasks have already been performed, so he has to *monitor* the analysis. The user may ask the expert, why he should do something, so the expert has to be able to *reason* about the advise he gives. Last but not least the user has to be able to achieve with help of the expert a result which is *superior* to the result which he would get in a comparable time without the expert.

These are exactly the characteristics of the prototype expert system which shall be suggested in this paper. Clearly, such a system is more than a 'user-friendly' program, because usually traditional programs have hardly any part of 'self-explanation' or knowledge about their own optimal use, 'knowledge' and its application is not seperated and there is seldom any guaranty about achieving optimal or nearly optimal results.

It will be, of course, an enormeous task to create an expert system with the above mentioned characteristics for a complete finite element analysis, starting from the right choice of mathematical models and material properties to the final dimensioning or design of the underlying structures. So we will restrict ourselves to the basic principles in the relatively small subtask of the design of optimal meshes and an optimal use of the p-version of the finite element method.

How does a CAD expert work ? First, he has to set up a clearly defined *goal*. In our context, the goal will be to perform a finite element computation on an optimal combination of element degree and mesh , i.e. the highest possible accuracy should be obtained for a certain amount of computer time and storage. To achieve the final goal, it has to be split into *subgoals*, each of which will be treated in the same way as the main goal. Finally, a *terminal subgoal* can be achieved by performing a *task*. This can be either the application of a certain algorithm which we will call *external task* or the answer to a question, e.g. "Are there reentrant corners in the domain of computation ?". These tasks will be called *internal tasks*.

External tasks can have the logical status 'TRUE', if they have been performed correctly, 'FALSE', if they cannot be performed and 'UNKNOWN', if they have not yet been at-

tempted. The answer 'YES' or 'NO' to an internal task causes the definition of new subgoals or the change of the logical status of other subgoals.

Thus a *tree of subgoals* is created, which defines the sequence of tasks which have to be performed under problem-specific conditions to achieve the final goal.

The expert-user-interface and the basic data structure

In this section we will describe the rule-based system which handles the 'surface knowledge' of the problem. The 'deep knowledge' will be explained in the section about the optimal mesh design in the hp-version of the finite element method.

A user can select from several options in his communication with the expert system . The more important features are:

STATUS : He can acquire the status of a special project, i.e. he can find out at any time, which subgoals have already been achieved.

ADVISE : He can ask the expert system, which task should be performed next.

WHY : He can ask, why he should perform a special task. He gets recursively the list of subgoals up to the root (final goal). An example is given in the final section of this paper.

The possibility to ask the question 'WHY' should not only be seen as a convenient way to 'educate' the user. More than that it is a possibility to 'discipline' the domain expert to justify his rules in detail. This justification should even go beyond the rule base itself to a *rule manual*, where a final explanation of the rules is given. The appendix of this paper should be seen as our rule manual for the optimal mesh design in the hp-version.

The system is running on a multi-process microcomputer (APOLLO DN420), one process executing the inference engine , the other process being the (algorithmic) application program. A typical outline of the screen can be seen in figure (4) of the final section.

The expert system needs two data structures, the *rule base* for the problem, i.e. the list of rules to achieve the final goal, and a *project base* where the status (TRUE, FALSE, UNKNOWN) of all tasks and goals of a special project at any time is stored.

Rules in the rule base have the following structure:

GOAL <nr> : <description of subgoal>

HAS INITIALLY THE LOGICAL STATUS : <true — false — unknown>

AND IS TRUE IF : <subgoal nr> AND—OR <subgoal nr> ... AND—OR <subgoal nr> IS TRUE

Internal tasks are defined as follows :

IF <question> IS ANSWERED WITH 'NO'

THEN <subgoal nr> GETS STATUS <true—false—unknown>

.

.

.

<subgoal nr> GETS STATUS <true—false—unknown>

The project base can be changed from two sides:

a) From the inference engine itself: The inference engine tries to change the status of all subgoals from 'UNKNOWN' to 'TRUE' or 'FALSE' by performing logical operations to infer the status of a father-node from the status of its subgoals. Furthermore, an internal task can change the status file .

b) From algorithms to be used in the execution of the whole system: An algorithm, i.e. an external task in the solution process can change the status in order to give the message to the inference engine about the (successful) execution of the algorithm. This communication can be achieved with only minor changes in the application programs (i.e.a subroutine-call).

The structure of this rule based system is of course very simple compared to large scale expert systems. Yet, even this simple system proves to be very effective due to the detailed and specific rules of the 'deep knowledge' which shall be descibed in the following.

THE RULES FOR AN OPTIMAL MESH DESIGN IN THE HP-VERSION

The successful design of an expert system hinges on a proper formulation of rules, principles, heuristics, experience (in general expertise) which can be incorporated into the knowledge based expert system frame. We will focus in this section on the basic principles of an optimal mesh design for the finite element analysis. For a summary of the mathematical foundation we refer to /4/ and the appendix.

First of all , the expert system has to determine whether a special problem belongs to its field of applications. We will restrict in this paper to *linear, two-dimensional elasticity problems on polygonal domains* although a generalization of our approach to more complicated cases is possible.

The finite element analysis based on our approach is performed in the following steps :

1. Interactive graphical mesh generation of a very crude *basic mesh*, which describes only the geometry of the problem.

2. Decision about the *critical* and *noncritical* elements of the basic mesh. Noncritical elements are those, where exponential rate of convergence can be achieved only by increasing the polynomial degree, critical are those, where mesh-refinement is necessary.

3. Extraction of critical data about the solution from a computation on an *elementary mesh* which can easily be constructed from the basic mesh. Critical data are, for example, stress intensity factors and strengthes of the singularities at reentrant corners. These data can be computed cheaply with sufficient accuracy using low p-degrees.

4. Prediction of the performance of various optimal mesh and degree combinations using the extracted critical data of step 3.

5. Graphical presentation of the predicted performance to the user. He can now decide on an optimal combination of mesh and polynomial degree which fits best to the desired accuracy and available resources of computer time and storage.

6. Mesh refinement for the final computation.

7. Finite element computation and assessment of the reliability of the computed data on the optimally refined mesh.

8. Presentation of the basic results to the user. Postprocessing, if the results are accepted; using the results analogously to step 4, if the results are rejected, because the original prediction was not accurate enough.

In contrast to adaptive finite element codes the final mesh is constructed in only *one step* after the preliminary computation in the basic mesh. Therefore, only minor additional effort for mesh-construction is necessary. In practical examples the computation on the final mesh took between 80 and 90 % of the overall computation, thus the basic computation and error prediction being nearly costless.

One basic idea is that a very crude basic mesh is classified into critical and noncritical areas (step 2). As the ultimate goal is to achieve highest accuracy for lowest computational cost, it is desired to perform an extension process (see /5/) which yields the optimal exponential rate of convergence in some error norm, in our case in the energy norm. As the p-version converges for smooth exact solutions with the desired exponential rate, a mesh refinement in smooth areas, i.e. far from reentrant corners or points of change of boundary conditions is not necessary. Moreover it turns out that in most engineering cases the accuracy of the p-version in noncritical areas is so high that the error there can be neglected compared to the overall error. Therefore the error prediction (step 4) considers only critical elements.

It is assumed that the exact solution there has essentially a 1-dimensional character. This principle is justified by well known results from the theory of partial differential equations For example , the solution near a reentrant corner behaves similar to Kr^{α} , where K is the stress intensity factor, r the distance and α the strength of the singularity . Therefore an auxiliary one-dimensional problem can be analyzed with extensive theoretical results being available in /6/. It has been shown there, that it is optimal to refine the mesh geometrically towards the singular point and increase the polynomial degree p simultaneously. The optimal relation between the number of refinement steps and the degree p depends on the strength of the singularity and can be computed explicitly. It can be proven that the optimal geometric progression factor in the hp-extension is .15 yielding a very strong refinement towards the singularity. Moreover it is possible to give an explicit functional relating the polynomial degree p, the number of refinement layers n and the strength of the singularity to the error in the energy norm.

This 1-D-analysis can be modified to be applicable to the two-dimensional case (see appendix and /4/) yielding the possibility to predict the error in a critical area for all combinations of geometric mesh refinement and polynomial degree p. This error prediction needs the stress intensity factors and the strength of the singularity as input data , yet these quantities can be computed cheaply with sufficient accuracy on the crude elementary mesh in step 3.

Moreover, if stress intensity factors and strengthes of *all* singularities in the exact solution are approximately known, a simple optimization procedure can predict optimal combination of polynomial degree and refinement at the various singularities for each desired number of degrees of freedom. The goal-functional for the optimization is the predicted error, variables are the polynomial degree p and the number of refinement layers n at the singularities $i = 1, ...s$. The number of degrees of freedom which is related to a mesh-degree combination is restricted to some upper bound N_0 .

In practice, not only one but a sequence of optimization processes is performed with the bound N_0 ranging in several steps from $NMIN$ to $NMAX$ thus showing the user a variety of optimal mesh-degree combinations and the related cost and predicted error.

AN EXAMPLE

A communication with the expert system

In the first part of this section we will show an example of a communication between the expert system and a user. In the second part the 'superiority' of the results is shown and thus the 'expertise' of the system is demonstrated. The computational results of this second part are taken from /4/.

Figure (1) shows the domain of computation and the loads of the model problem. The domain has 9 reentrant corners (including the two tips of the crack interior to the domain), constant traction is applied along the edge CD and symmetry boundary condition is imposed along AB . Isotropic material with $E = 3.0e8$ psi and $\nu = .3$ was assumed.

Figure (2) shows the basic mesh of this problem as it was constructed interactively in the expert system using a modification of a meshgenerator of MODULEF /7/. The basic mesh is as coarse as possible, only modelling the geometry of the problem.

In the next step of the analysis the decision about critical and noncritical elements is made. Each element is critical, because each has at least one reentant corner as a node.

The next step extracts critical data about the solution. As has been shown, the stress intensity factors K and exponents α of the singular functions provide means to predict the error of the energy norm for all combinations of (geometric) mesh and polynomial degree p. For a sufficiently accurate extraction of stress intensity factors it is necessary to have a mesh such that at most one singular point is in each element. Therefore out of the basic mesh an *elementary mesh* is constructed automatically (figure (3)) and stress intensity factors are computed using the extraction procedure described in /8/.

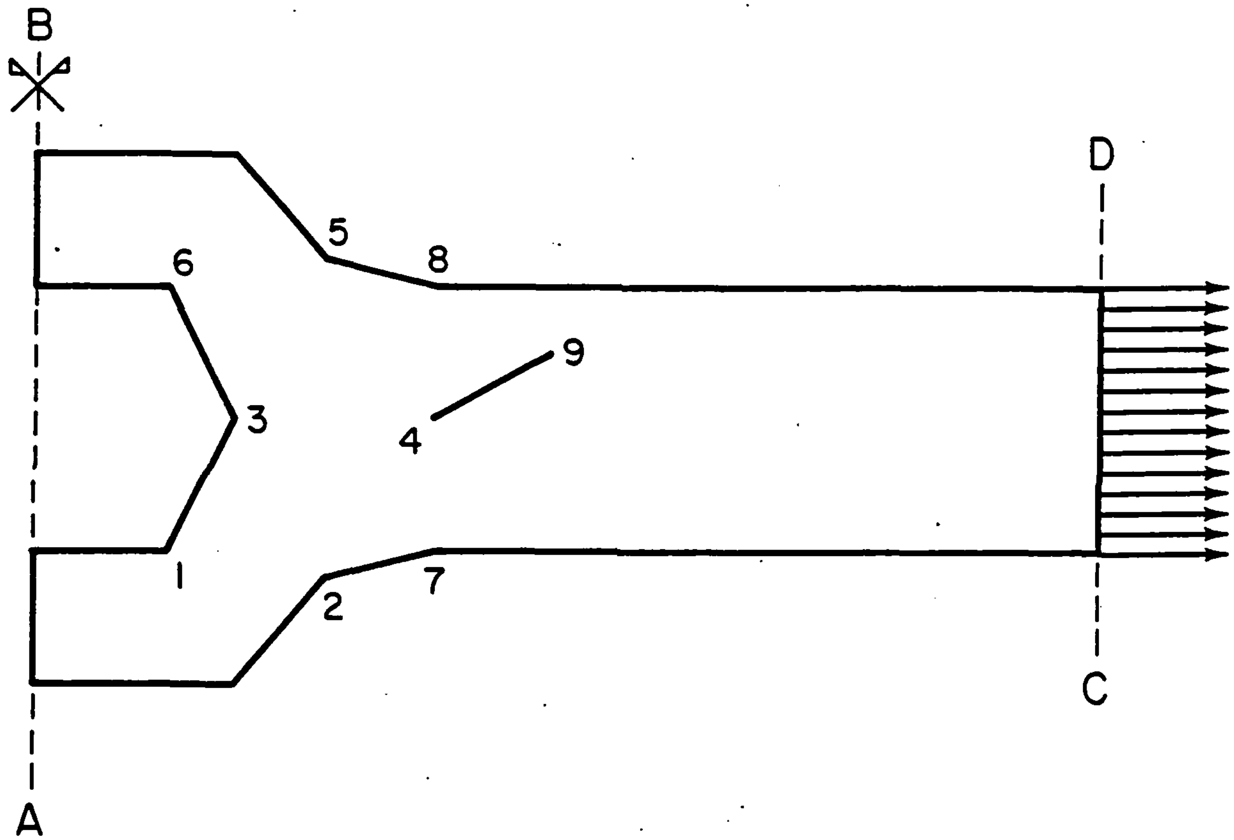


figure 1 : Domain and loads for model problem

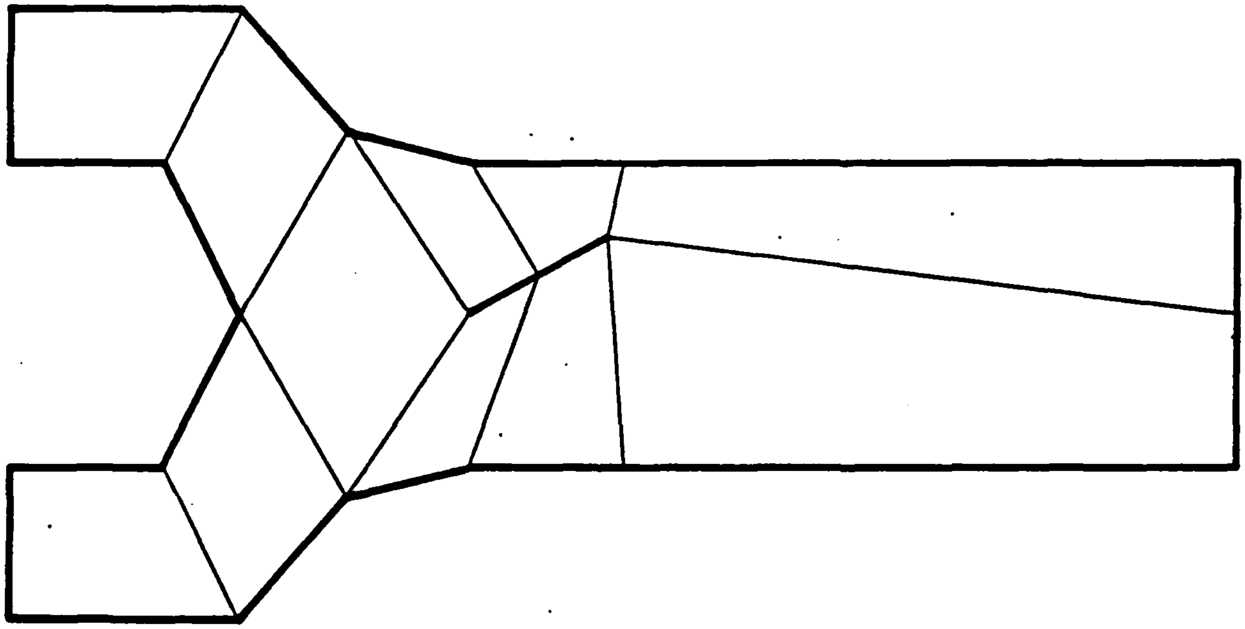


figure 2 : Basic mesh for model problem

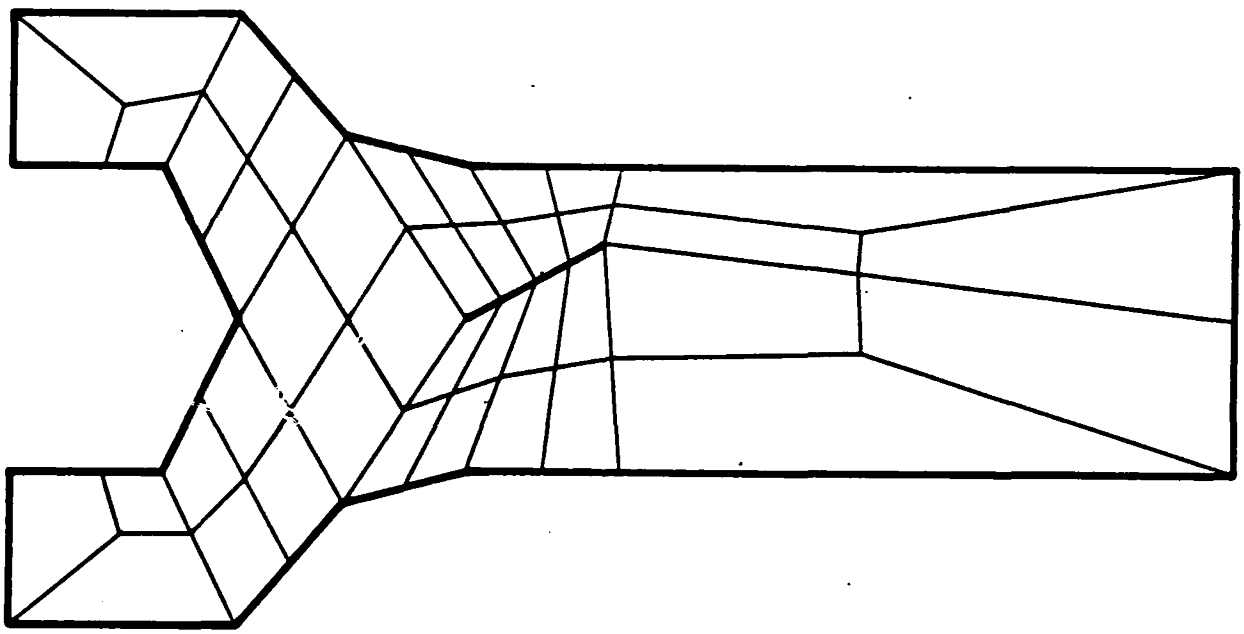


figure 3 : Elementary mesh for crude computation

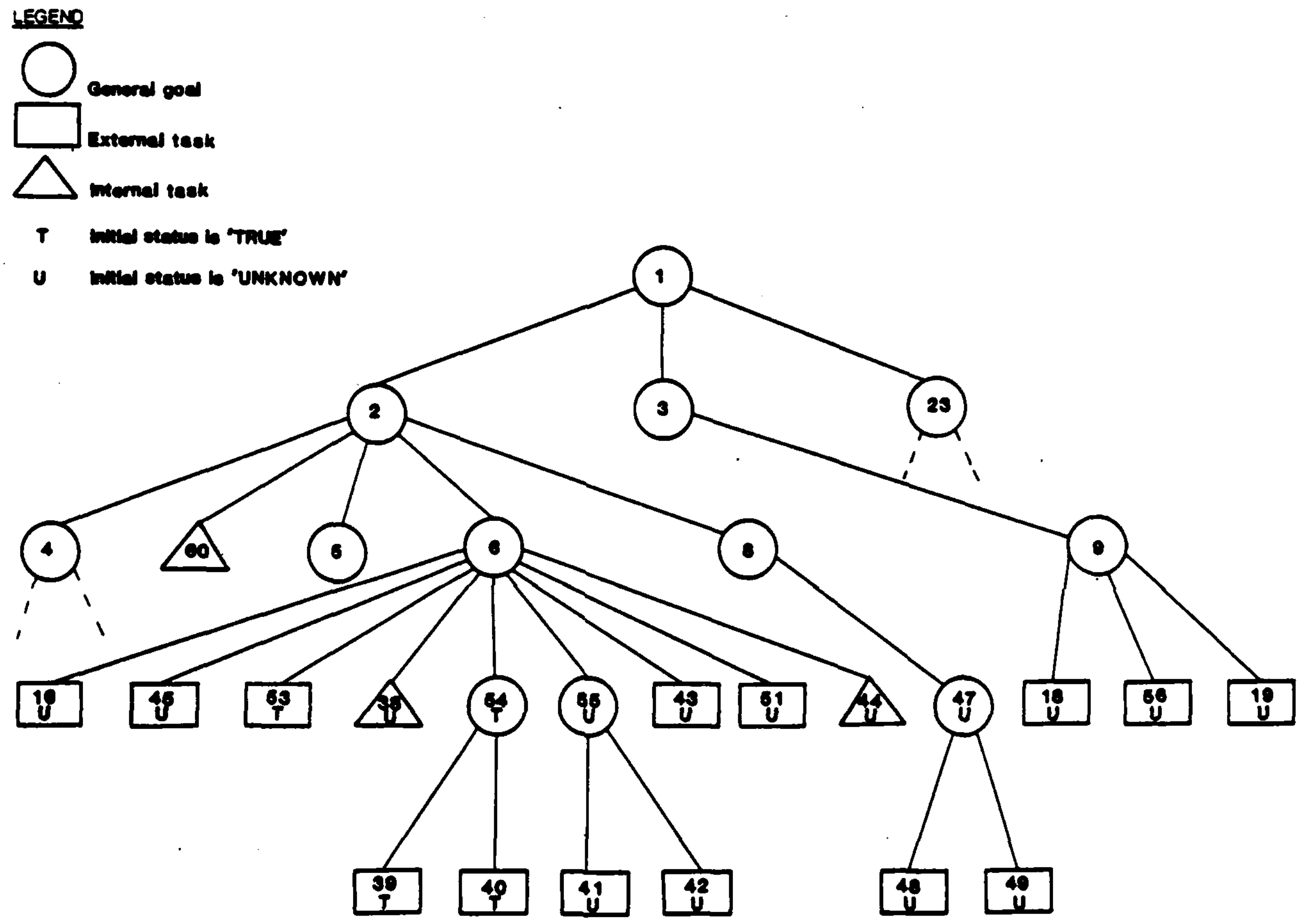


figure 4 : Subtree corresponding to error prediction

Next the error for various combinations of mesh and polynomial degree can be predicted and the mesh can be refined for the final computation. In the following a typical communication between the user and the expert during these two steps is shown. The example covers steps 4 and 5 of the eight steps described in the section on the rules for the optimal mesh design. After each advise of the expert the user performs the indicated steps in the analysis, the successful execution of the algorithmic programs being directly reported to the expert system via the status file. This part of the analysis is covered by the subset of the goal-tree as shown in figure (4). The description part of the rule base which corresponds to the rules invoked in this subtree is given in the following.

T1 : perform hp-computation with optimal accuracy

T2 : generate a final mesh

T3 : perform a p-version FE-computation on final mesh

T4 : generate an elementary mesh

T5 : extract critical data

T6 : predict optimal combinations of mesh and p-degree

.

.

.

T8 : make decision about final mesh

T9 : refine elementary mesh

.

.

.

T16 : start PREDICT_ERROR

.

.

.

T18 : move curser to left working pad, type EXIT , 'pop' window and start

REFINE_ELEMENTARY_MESH

T19 : define loads, constraints and material properties

.

.

.

I38 : Do you want to achieve a high accuracy solution

T39 : give range of polynomial degree (rule of thumb : it should be 2 to 4)

T40 : choose NMIN=80 * NCORN , NMAX = 150 * NCORN (NCORN is number of reentrant corners)

T41 : give range of polynomial degree (rule of thumb : it should be 3 to 6)

T42 : choose NMIN=150 * NCORN , NMAX = 500 * NCORN (NCORN is number of reentrant corners)

T43 : provide the increment in the degrees of freedom (10 steps from NMIN to NMAX is reasonable)

I44 : Is the desired accuracy in the predicted range

T45 : type EXIT, 'pop' window and start PREDICT_ERROR again

.

.

.

T47 : find out optimal combination of p-degree and mesh

T48 : move curser to right working pad and select a polynomial degree on the lower left envelope of the error curves

T49 : select NDOF, which correspond to a favourable point. (NOTE : computational cost grows about as NDOF**2 !)

.

.

.

T51 : press key F5 and type DISPLAY_PREDICTION (tell, when DONE)

.

.

.

T53 : adjust range of p-degrees and degrees of freedom

T54 : predict error for solutions with moderate accuracy (error in energy norm : 5 - 10 %)

T55 : predict error for solutions with high accuracy (error in energy norm : 1 - 5 %)

T56 : refine at each reentrant corner geometrically with the number of layers specified in PREDICT_ERROR

T59 : refine mesh,choose SEPERATE SINGULARITIES (1) and move curser subsequently to all reentrant corners

I60 : Do you want to use the elementary mesh for your final computation

.

.

.

The inference engine displays after each request for an 'advise' the terminal task with status 'UNKNOWN' farthest to the right in the goal tree. So the next terminal task to be performed will be task 16:

Expert : YOU CAN SELECT ONE OF THE FOLLOWING OPTIONS :

STATUS , ADVISE , WHY , END , HELP , DONE

User : ADVISE

Expert : Next task is T16 : start PREDICT_ERROR

The user starts the error prediction module PREDICT_ERROR. This module gives the message to the expert system that it has been started, i.e. the status of task 16 is changed to 'TRUE'.

User : AD

Expert : Do you want to achieve a high accuracy solution (YES - NO) ?

User : YES

This internal task is an example of how 'hidden' branches can be opened and open branches can be closed. The answer to this question invokes several rules of thumb, which help to find reasonable input data for the optimization program. The answer 'YES' leaves the status of all subtasks as it is, so the next subtask will be T41. An answer 'NO' would change the status of T54, T39 and T40 to 'UNKNOWN' and of T55 , T41 and T42 to 'TRUE'. Thus the next subtask would be T39, a rule of thumb for 'moderate accuracy solutions'.

User : AD

Expert : Next task is T41 : give range of polynomial degree (rule of thumb : range from 3 to 6)

The optimization program needs the sequence of polynomial degrees and the sequence of maximally allowed number of degrees of freedom , i.e. N_0 in the formulation of the optimization problem. The following communication sequence shows the 'reasoning'-capability of the expert system, which explains, why it gives some advise. This reasoning is constructed from the sequence of father-goals starting with the next task, i.e T42 up to the final goal.

User : AD

Expert : Next task is T42 : choose NMIN=150 * NCORN , NMAX = 500 * NCORN (NCORN is number of reentrant corners)

User : WHY

Expert : T42 : choose NMIN=150 * NCORN , NMAX = 500 * NCORN (NCORN is number of reentrant corners) is necessary to predict error for solutions with high accuracy (error in energy norm : 1 - 5%)

User : WHY

Expert : T55 : predict error for solutions with high accuracy (error in energy norm : 1 - 5%) is necessary to predict optimal combinations of mesh and p-degree

User : WHY

Expert : T6 : predict optimal combinations of mesh and p-degree is necessary to generate a final mesh

User : WHY

Expert : T2 : generate a final mesh is necessary to perform hp-computation with optimal accuracy

User : WHY

Expert : T1 : perform hp-computation with optimal accuracy is the final goal

The user chooses now NMIN = 1500 , NMAX = 4000 and an increment of NDELT = 400 and gets after the optimization a plotted output of the prediction curves. This state is shown in figure (5).

User : AD

Expert : Is the desired accuracy in the predicted range (YES - NO) ?

The prediction module does not only compute *one* optimal combination of polynomial degree and mesh but it shows the user a *variety* of possibile optimal choices, thus giving him many options for his final decisions on the mesh construction. The user will answer this question with 'YES', because in our model problem the rules of thumb on the choice of polynomial degree and the range of degrees of freedom were right. An answer 'NO' would change the status of subtasks T45, T43, and T51 to T53 to 'UNKNOWN' thus entering a 'loop' to adjust the input data which led to the rejected result. With the acceptance of the predicted error the next step is to inquire about the mesh which corresponds to the desired combination of error and cost of the computation.

Expert : Next task is T48 : select a polynomial degree on the lower left envelope of the error curves

User : AD

Expert : Next task is T49 : select NDOF (NOTE : computational cost grows about as NDOF**2 !)

The next step is now to enter the interactive refinement-module and refine the mesh according to the advise given by the error prediction. This could, of course, be performed completely automatically, but it seems favourable to leave the user the choice to modify the suggested mesh design. He may, for example, not be interested in the solution in areas near some of the singularities, thus refineing there less than suggested by the error prediction, which always tries to ballance the error in the whole domain of computation. The local refinement is performed simply by providing the node number of the refinement center and the number of element layers to be created there. The mesh after refinement is shown in figure (6).

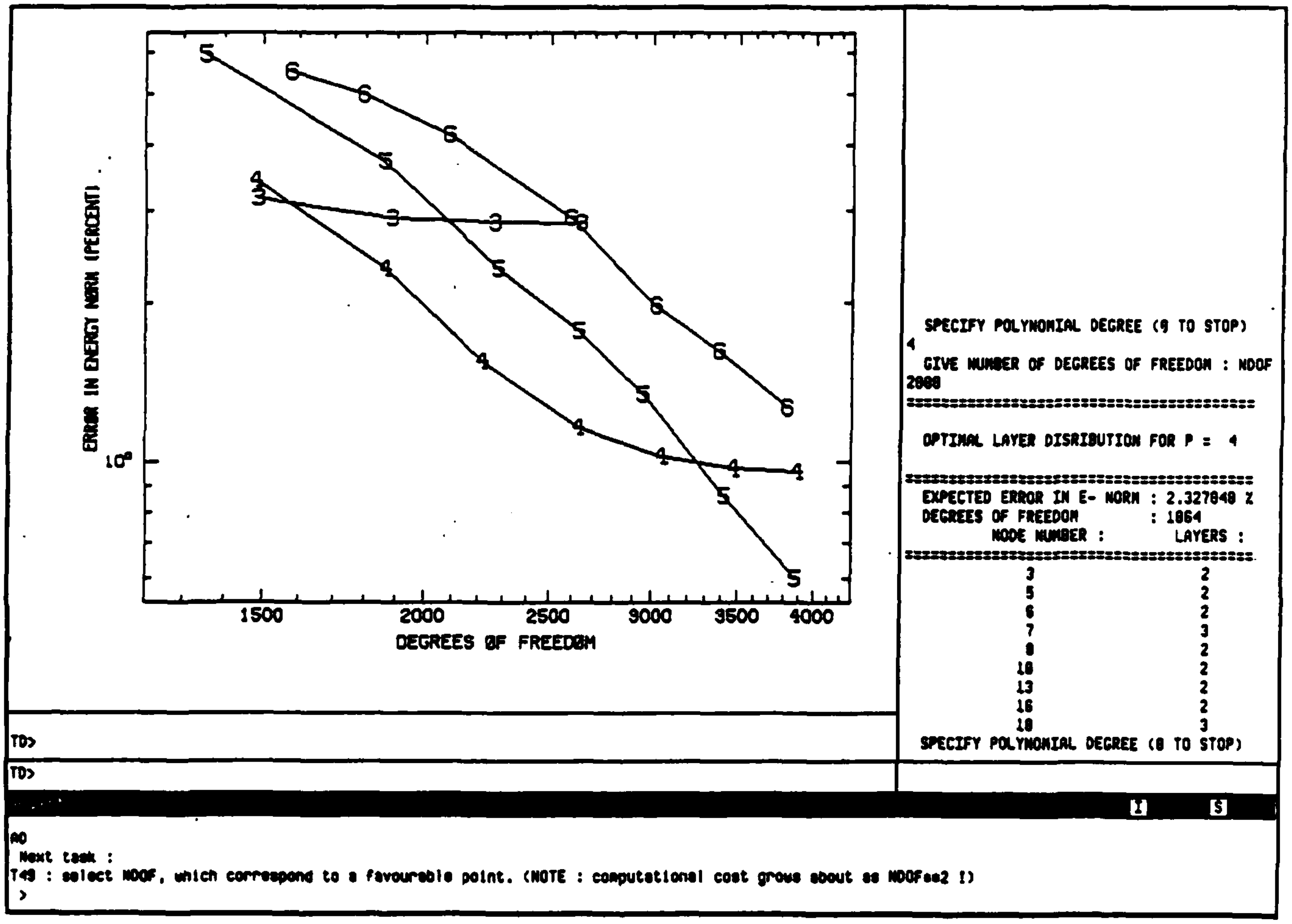


figure 5 : Screen layout after error prediction for polynomial degrees p from 3 to 6, N_0 = 1500 to 4000 in steps of 400

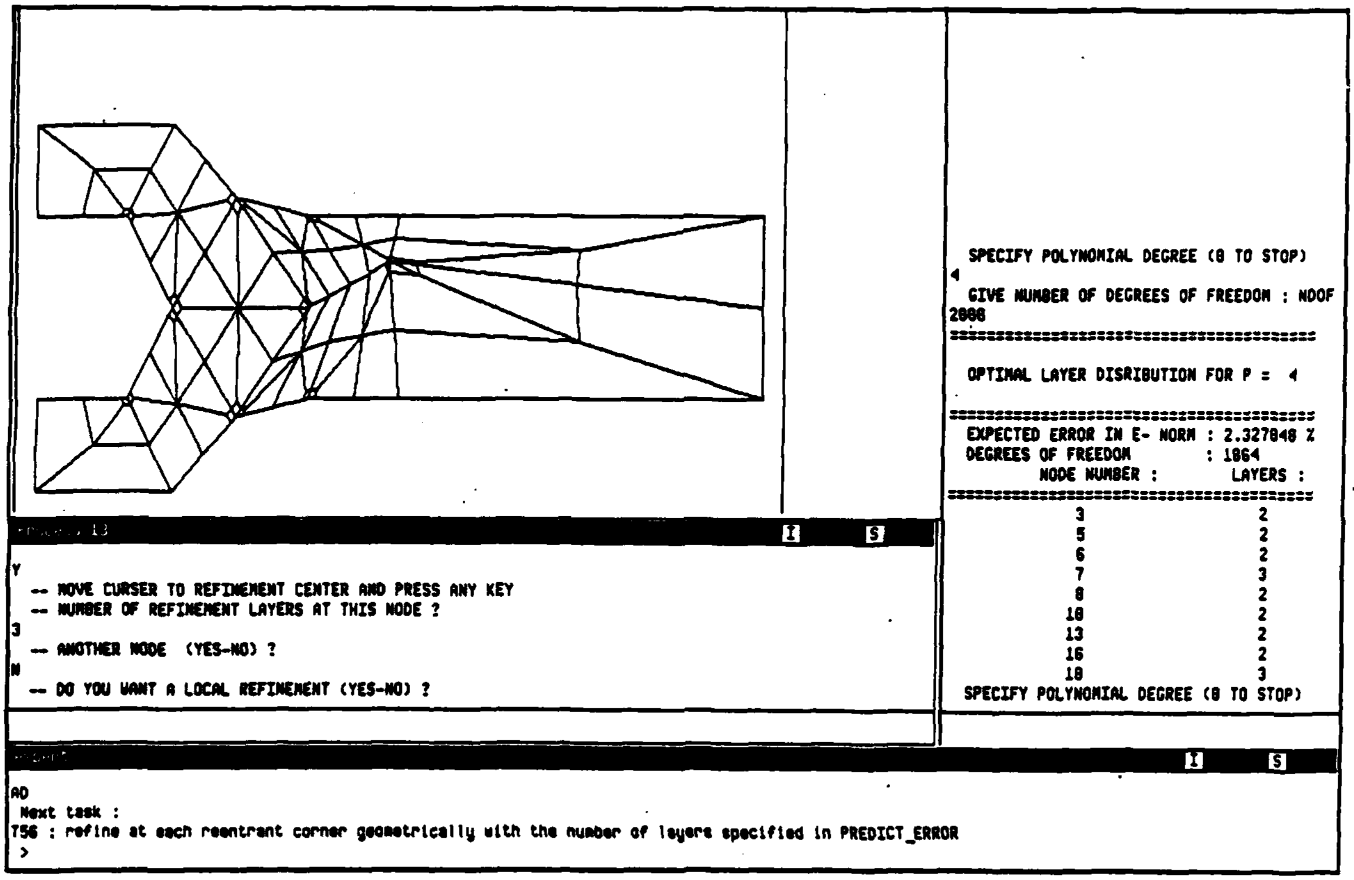


figure 6 : Mesh after refinement

Now the loads, boundary conditions and material properties can be specified and the final computation can be performed.

Numerical results

In a practical application of the hp-expert system only one computation on a final mesh has to be performed, assuming that the error prediction estimates the true error reasonable accurate. In order to show the behaviour of the system in a wide range of combinations of computer cost and accuracy, , we performed here finite element computations for all meshes and degrees as specified in an error prediction for p ranging from 2 to 5 and the number of degrees of freedom ranging from 400 to 4000. The predicted error is shown in figure (7). The exact energy was estimated by extrapolation on an extremly refined mesh and high polynomial degree to U =0.0367809. Figure (8) shows the exact error in the energy norm for all computed combinations of mesh and polynomial degree. Comparing figures (7) and (8), it can be seen that the real error for a certain p-degree levels off earlier than the predicted error. This seems to be due to the fact, that the error predictor neglects the error in 'smooth' parts of the solution , which are still significant for low p-degrees.

However 8 of 10 optimal points (points at the lower left envelope in figure (7) which are marked by circles in figure (8)) turn out to be really the best meshes for the specified number of degrees of freedom, the remaining two meshes having only mildly larger error than the actual best combinations. It should be mentioned, that the optimal meshes constructed in the feedback process, converge at an exponential rate , which shows, that the feedback really yields optimal mesh design.

By this feedback process it is possible to construct a mesh, which yields an accuracy in the energy norm of about 2% for 3000 degrees of freedom. Extrapolation from the elementary mesh shows, that this accuracy could be obtained with a quasiuniform mesh using linear elements (assumed convergence rate $\delta = .25$) with about 1.e7 degrees of freedom. An adaptively constructed mesh, using the h-version with linear elements (convergence rate $\delta = .5$) should yield this accuracy with about 34,000 degrees of freedom. About the same amount of dofs would be necessary for a pure p-version on the elementary mesh. From these estimations it can be seen, that about one order of magnitude in the number of degrees of freedom is gained compared to an adaptive h-version or a pure p-version. So roughly two orders of magnitude in storage and computational time is gained, if high accuracy solutions with an error in the range of 1 to 2% are desired.

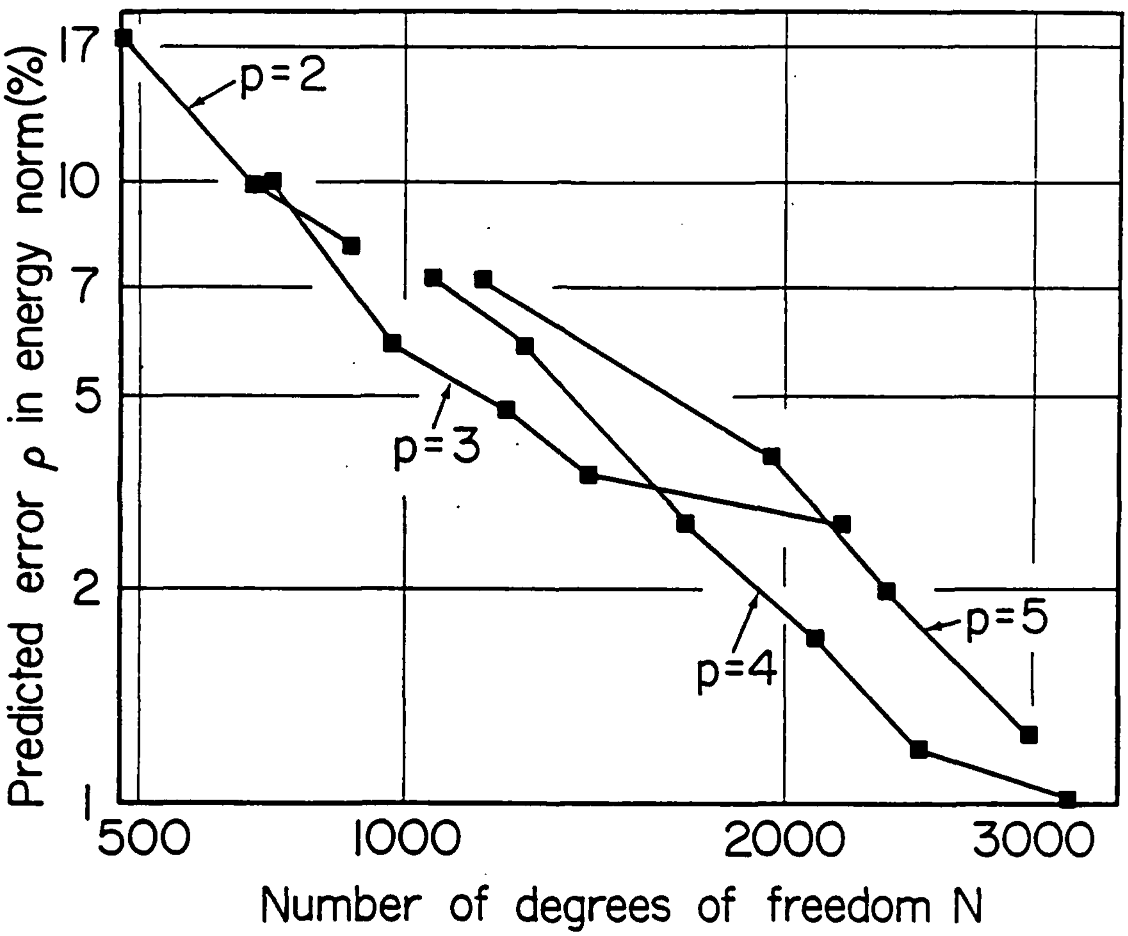


figure 7 : Predicted error for model problem

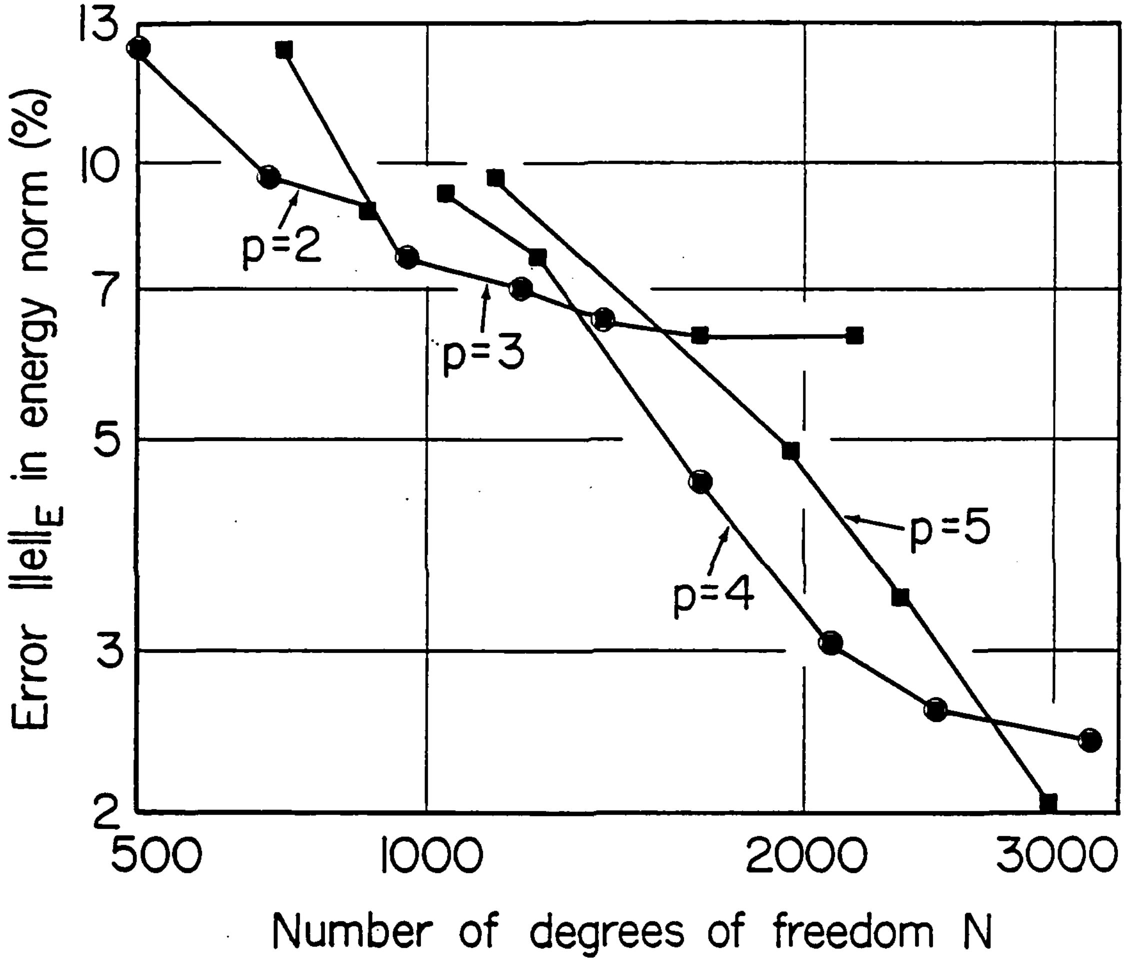


figure 8 : Actual error for model problem

The additional cost for the feedback is negligable. On an Apollo 420 the overall time for the computation on the elementary mesh ($p = 1,2,3$) together with graphical mesh generation, the sequence of optimizations for $p = 2,3,4,5$, N_0 from 400 to 4000 in steps of 200 degrees of freedom and the construction of a refined mesh with 2985 dofs was less than 800 CPU sec, whereas the final computation on the refined mesh with $p = 5$ took more than 4000 cpu seconds.

Moreover, it should be mentioned, that the human time is completely independent of the desired accuracy, a situation, which is totally different from conventional finite element analysis, where higher accuracy can only be achieved by time consuming construction of refined meshes. Only the basic mesh has to be constructed by human interferance and after this , only decisions about the progress of the analysis, guided by the advise of the expert system have to be made.

SUMMARY

The hp-version of the finite element method is a newly developped technique to achieve high accuracy in the results at low cost of computational time and storage requirement. Combining local mesh refinement and increase of the polynomial degree it shows exponential rate of convergence even in the presence of singularities in the exact solution. It has been shown in this paper, that a simple expert system with mathematical knowledge and heuristic rules for the optimal mesh design provide a frame to use the hp-version for practical engineering problems. The expert system monitors the progress of the analysis, guides the user through the various steps and is able to reason about its own advise. It is thus a prototype for a more general expert system which should help the user throughout a complete finite element analysis.

APPENDIX : SOME MATHEMATICAL PRINCIPLES OF THE HP-VERSION

The crutial step in the feedback analysis is the prediction of the error (step 4). This is the 'deep knowledge' which shall be described in this appendix.

Let us consider the plain (strain) elasticity problem on a polygonal domain Ω with homogenious material, elasticity module E and Poissons ratio $0 \leq \nu < 1/2$. The vertices of Ω shall be denoted by A_i , the internal angle at A_i by ω_i .

For simplicity, we will assume, that no body forces are present. In a neighborhood S_i of the vertices A_i the exact solution of the problem can be written as

$$u_0 = \sum_{j=1}^{Q_i} K_{ij}\Phi_{ij}(r_i)g_{ij}(\theta_i) + w_{Q_i,i} \tag{1}$$

where (r_i, θ_i) are polar coordinates with respect to A_i , $\Phi_{ij}(r_i),, g_{ij}(\theta_i)$ are a-priori known functions independent of u_0 (depending on ω), K_{ij} (scalars) are stress intensity factors (dependent on u_0) and $w_{Q_i,i}$ is an analytic function on $\overline{S_i}\backslash A_i$ which is smoother than the first term on the right hand side of (1). Q_i is a positive integer which will be chosen for our practical purposes as 1 or 2. For more details, see /9/,/10/, /11/.

The function $g_{ij}(\theta_i)$ is analytic up to the boundary of S_i and $\Phi_{ij}(r_i) = Re(r_i^{\alpha_{ij}} log^{\rho}(r_i)).\alpha_{ij}$ are in general complex numbers with positive real parts and $Re(\alpha_{i,j+1}) \geq Re(\alpha_{ij})$. Hence the first terms in (1) are the most singular ones. For the way to compute stress intensity factors K_{ij} we refer to /8/.

Let us now assume, that the polynomial degree of all the elements is the same. It has been shown /12/, that under assumptions, which are always satisfied in engineering problems, for a proper mesh $\Delta(p)$ depending on p the error of the finite element solution u_{FE} measured in the energy norm decays exponentially. More precisely

$$\|e\|_E = \|u_{FE} - u_0\|_E \leq e^{-aN^{1/3}(p)} \quad ; \quad p \longrightarrow \infty \tag{2}$$

where $a > 0$ and N is the number of degrees of freedom.

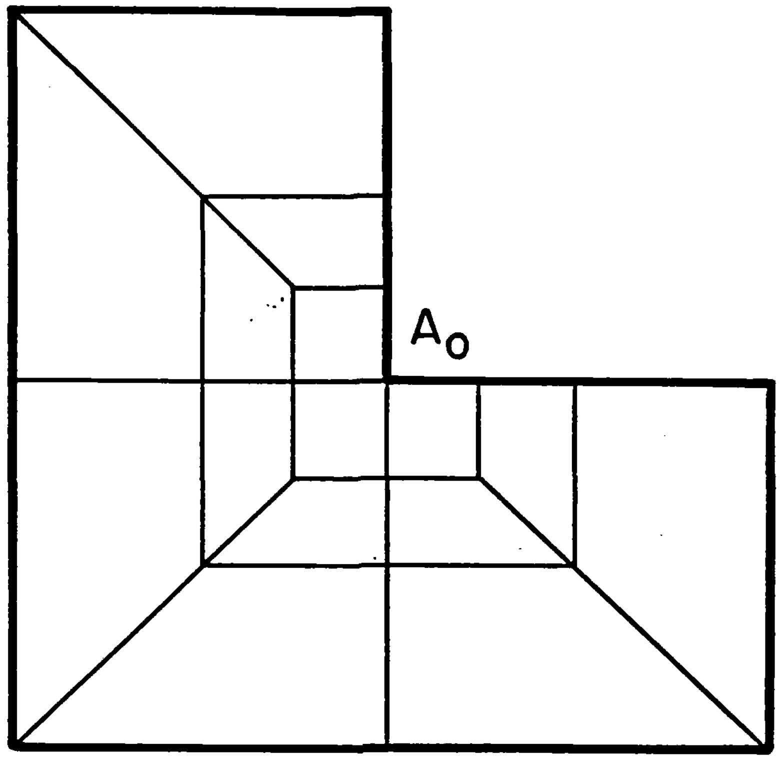


figure 9 : Example of a geometric mesh with two refinement layers

The error in particular elements has two essential parts. The *local* one depending only on the solution in the given element and the *global* one which reflects the influence of the error in the entire domain (so called pollution error). The global error is (for properly designed meshes) smaller than the local one. If an element is not adjacent to any vertex, then the local error of the (p-version) finite element method is of the order

$$e^{-aN_1^{1/2}}$$

where N_1 is the number of degrees of freedom associated to the element whereas the error is of order $N_1^{-\beta}$ when the particular element has a vertex in a corner of the domain. Hence the decay of the error is more rapid in areas far from corners of the domain than in the neighborhood of the corners. Therefore only proper refinement in the neighborhood of corners is needed while in the elements not adjacent to corners high accuracy can be achieved only by increasing p. For practical engineering accuracies we can treat convex corners as no corners (provided that the boundary condition is the same on both sides of the vertex).

So the proper mesh $\Delta(p)$ is geometric with refinement only in the neighborhood of every reentrant corner. An example of such a mesh with 2 layers at A_0 is shown in figure (9).

It has been shown, that the optimal number n of layers of elements increases with the polynomial degree p, that the optimal ratio of the geometric mesh is independent of the strength of the singularity α, p and the number n of layers and has a magnitude of .15 (see /6/,/12/). Thus a very strong grading towards the singularity is obtained.

If the mesh is fixed (with different numbers of layers) and the polynomial degree p increases, then the error behaves as schematically shown in figure (10). For each fixed number of layers a typical reverted S-curved behaviour of the error is observed. We see the preasymptotic phase (curved down), when the error decays exponentially and the asymptotic phase, when the error decays algebraically (straight line). For theoretical results we refer to /6/,/8/.

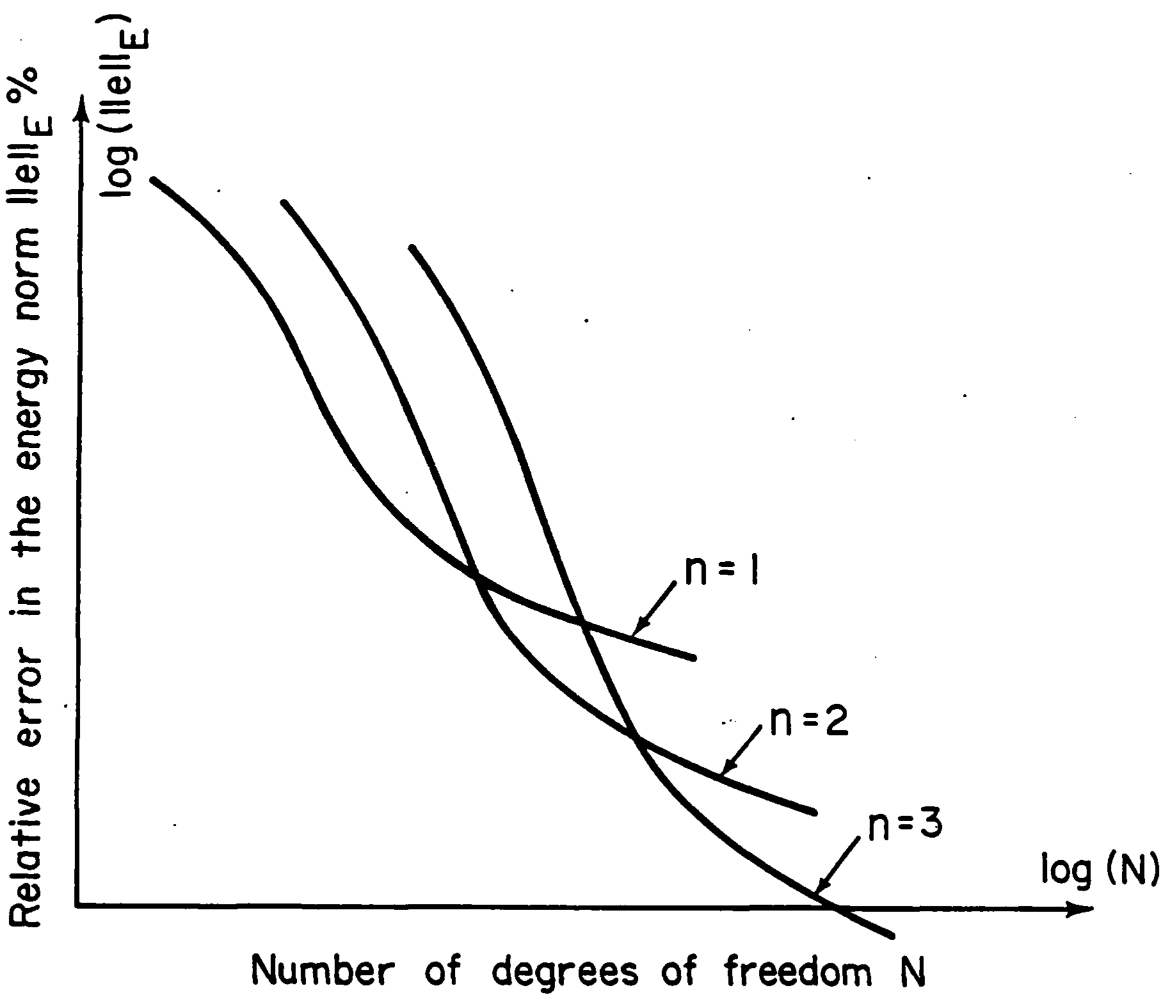


figure 10 : $\|e\|_E$ in dependence of the number of degrees of freedom for different number of layers n

Let us now investigate the error in critical elements, i.e. elements adjacent to reentrant corners, in detail. Because the function g_{ij} in (1) is smooth, the error behaviour near A_i is qualitively the same as if $g_{ij} = 1$. Hence, the error will be essentially the same as in a one-dimensional setting in the interval $(0, B)$ with the weight x expressing polar coordinates. Yet, the hp-version in one dimension has been studied extensively /6/ and the error analysis can be adopted to the two dimensional case of our problem near the singular points /4/.

To show the basic principles, let Δ be the one dimensional mesh on $I = (0, B)$.

$$\Delta : o = x_0 < x_1 < x_2 < ... < x_M = B$$

where

$$x_i = Bq^{M-1} \quad , \quad q < 1$$

and denote

$$I_j = (x_{j-1}, x_j)$$

Assume that $f = x^\beta$ and define

$$E(\beta, p, q, M) = (\inf_\omega \int_0^\beta (f' - \omega')^2 x dx)^{1/2} \tag{3}$$

where the infimum is taken over all functions $\omega \in H^1(I)$, ω is a polynomial of degree p on $I_i, i = 1, ..., M$ and $\omega(x_i) = f(x_i)$.

We have now

Theorem :

$$E(\beta, p, q, M) = CB^\beta [\frac{q^{2\beta(M-1)}}{p^{4\beta}} + \frac{r^{2p}}{p^{2\beta}} \frac{(1 - q^{2\beta(M-1)})(1-q)q^{\beta-1}}{1-q^{2\beta}}]^{1/2} \tag{4}$$

$$r := \frac{1-\sqrt{q}}{1+\sqrt{q}}$$

where $D_1(\beta) \leq C \leq D_2(\beta)$ with D_i independent of M and p. (See /4/ for the proof).

Let us now return to the error in a critical element e_i having a vertex in the vertex A_i of the polygonal domain. (see figure (11)). We will assume, that the exact solution to be approximated is

$$K_{1i} r^{\alpha_{1i}} \Phi_{1i}(\theta) + K_{2i} r^{\alpha_{2i}} \Phi_{2i}(\theta)$$

Furthermore, let

$$\rho_i^2 := C\omega_i \sum_{j=1}^{2} K_{ji}^2 E^2(\alpha_{ji}, p, q, M) B_i^{\alpha_{ji}} \tag{5}$$

ω_i is the angle of the critical element at A_i and B_i is the radius of the smallest circle centered at A_i covering the element.

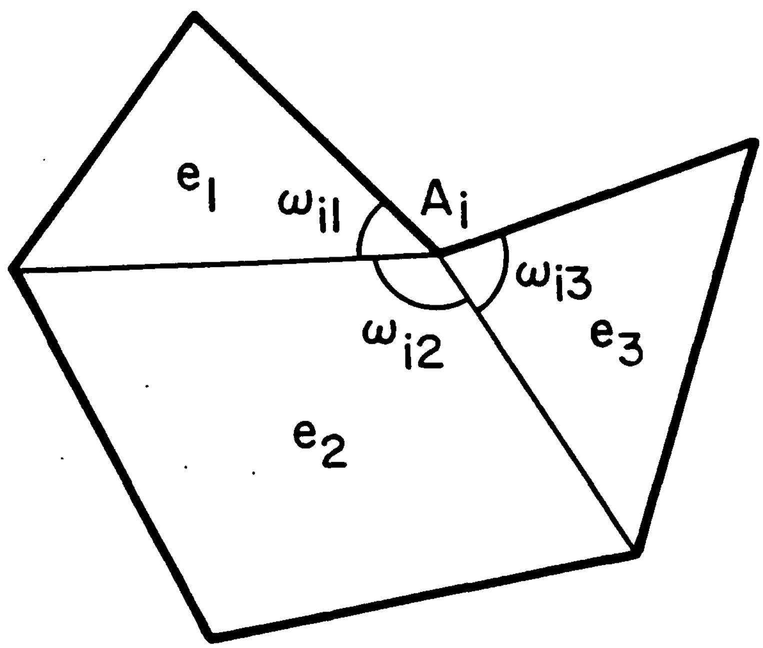


figure 11 : Scheme of elements at vertex A_i

Expression (5) is an error functional and will be called error predictor. It can be seen in the numerical example, that ρ_i reliably gives all characteristics of the error behaviour depending on M and p.

The value q will be chosen of the order .15 which is the optimal value for one dimensional problems.

As has been stated we will asume that the decisive areas for the accuracy are the critical elements which have to be refined. Hence we will assume that the only error is in these elements and that the stress intensity factors are known.

For given p and number of layers $n_i = M_i - 1$ in the $t(i)$ elements adjacent to the critical vertices $A_i, i = 1, ..., s$ we will use the error prediction

$$\rho^2(p, M_i) = C \sum_{i=1}^{s} \sum_{l=1}^{t(i)} \omega_{i,l} \sum_{j=1}^{2} K_{ji}^2 E^2(\alpha_{ji}, p, q, M_i) B_{il}^{\alpha_{ji}} \tag{6}$$

where K_{ji} are the (known) stress intensity factors in the vertex A_i and α_{ji} the exponents of the singular functions in (1).

The number of degrees of freedom over all the refined mesh will be denoted by

$$N(p, M_i, i = 1, ..., s)$$

.

Assuming that the computational work is a function of N only, we can formulate the following optimization problem :

Given an upper bound N_0 of the total number of degrees of freedom find p and M_i so that it minimizes (6) under the constraint

$$N(p, M_i, i = 1, ..., s) \leq N_0$$

For practical purposes, p can be restricted to a maximum degree p_{max} and it is also reasonable to restrict the number of layers. A possible choice for the maximally allowed number of layers is $M_{max} = 2p_{max}$ as it was shown in /12/, that the optimal combination of M and p in the case of a single crack-tip singularity is given by $M = p$.

With these restrictions the optimization problem is finite and various methods for constrained, finite optimization could be used. Yet it turns out that (6) has often many local extrema so that the exact optimum will be hard to find. Nevertheless it is not necessary to find the minimum exactly because various simplifications in obtaining (6) have been

made. A simple heuristic algorithm, which has been described in /4/ gives in all our test examples nearly optimal results with only minor computational effort.

Finally, the constant C in (6) has to be estimated to get not only a qualitive but also a quantitive prediction of the error for different combinations of mesh and polynomial degree. This can be done by a *calibration* of the predicted error on the crude mesh for low p-degrees. For example we can compute the solutions for $p = 1$ and $p = 2$ or $p = 1, 2, 3$ on the elementary mesh and determine the approximate error by extrapolation, assuming that

$$\|e\|_E = CN^{-\delta} \tag{7}$$

for the p-version on a fixed mesh where δ depends on the smoothness of the solution. Experience shows, that for low p and coarse meshes, $\delta = .70$ to $\delta = .75$ gives reasonable results in most practical cases.

REFERENCES

/1/ Hayes-Roth,F., D. A. Waterman and D. B. Lenat (eds), Building Expert Systems, Addison-Wesley Publishing Company, Inc., 1983.

/2/ Fenves, S., A framework for a knowledge-based finite element analysis assistant, presented at WAM/ASME Symposium on Application of Knowledge-Based Systems to Engineering Analysis and Design, Miami Beach, Florida, Nov. 17-22, 1985.

/3/ Zumsteg, J.R. , Flaggs, D. L. , Knowledge-based analysis and design systems for aerospace structures, presented at WAM/ASME Symposium on Application of Knowledge-Based Systems to Engineering Analysis and Design, Miami Beach, Florida, Nov. 17-22, 1985.

/4/ Babuška,I. , Rank, E. , An expert-system-like feedback approach in the hp-version of the finite element method, Inst. for Phys. Sc. and Tech., Lab. for Num. Analysis, Technical Note BN-1048, 1986 .to appear in : Finite Elements in Analysis and Design.

/5/ Babuška,I., Feedback, adaptivity and a-posteriori estimates in finite elements: Aims, theory and experience. Proc. Int. Conf. on Accuracy Estimates and Adaptive Refinements in Finite Element Computations (ARFEC) , Lisbon, Portugal, 1984.

/6/ Gui,W., Babuška,I., The h, p and h-p versions of the finite element method in 1 dimension: Part 1 : The error analysis of the p-version. Part 2 : The error analysis of the h and h-p versions. Part 3 : The adaptive h-p version. Inst. for Phys. Sc. and Tech., Lab. for Num. Analysis, Technical Note BN-1036, BN-1037, and BN-1038, 1985. To appear in Numerische Mathematik.

/7/ Bernadou,M. , et al. , MODULEF, une bibliotheque modulaire d'elements finis. Institut National de Recherche en Informatique et en Automatique (INRIA), 1985.

/8/ Szabo, B. A., Babuška, I., Computation of the amplitudes of stress singular terms for cracks and reentrant corners, Technical Note WU/CCM-86/1 , Center for Computational Mechanics, Washington University, 1986.

/9/ Muskhelishvili, N. I., Some basic problems of the mathematical theory of elasticity, Published in Russian in 1933, English translation of the 3rd edition : P. Noordhoff, Ltd., Groningen, Holland, 1953.

/10/ Williams, M. L., Stress singularities resulting from various boundary conditions in angular corners of plates in extension, J. Appl. Mech. ASME, pp. 526-528, 1952.

/11/ Grisvard,P. , Behaviour of the solution of an elliptic boundary value problem in a polygon or polyhedral domain, in : Numerical Solution of Partial Differential Equations, 3rd ed., (B. Hubbard, Ed.), Academic Press, New York, pp. 207-274, 1976.

/12/ Guo, B., Babuška, I., The h-p version of the finite element method. Part I : The basic approximation results. Part II : General results and applications. Computational Mechanics, I, 1986.

The Laboratory for Numerical analysis is an integral part of the Institute for Physical Science and Technology of the University of Maryland, under the general administration of the Director, Institute for Physical Science and Technology. It has the following goals:

- To conduct research in the mathematical theory and computational implementation of numerical analysis and related topics, with emphasis on the numerical treatment of linear and nonlinear differential equations and problems in linear and nonlinear algebra.

- To help bridge gaps between computational directions in engineering, physics, etc., and those in the mathematical community.

- To provide a limited consulting service in all areas of numerical mathematics to the University as a whole, and also to government agencies and industries in the State of Maryland and the Washington Metropolitan area.

- To assist with the education of numerical analysts, especially at the postdoctoral level, in conjunction with the Interdisciplinary Applied Mathematics Program and the programs of the Mathematics and Computer Science Departments. This includes active collaboration with government agencies such as the National Bureau of Standards.

- To be an international center of study and research for foreign students in numerical mathematics who are supported by foreign governments or exchange agencies (Fulbright, etc.)

Further information may be obtained from Professor I. Babuska, Chairman, Laboratory for Numerical Analysis, Institute for Physical Science and Technology, University of Maryland, College Park, Maryland 20742.

END
11-86
DTIC